

**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

**BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☐ **FADED TEXT OR DRAWING**
- ☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKewed/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/775,862	02/01/2001	Chunguang Chris Liu	50325-0539	8413

29989 7590 08/31/2004

HICKMAN PALERMO TRUONG & BECKER, LLP  
1600 WILLOW STREET  
SAN JOSE, CA 95125

EXAMINER

PATEL, ASHOKKUMAR B

ART UNIT	PAPER NUMBER
----------	--------------

2154

DATE MAILED: 08/31/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

<b>Office Action Summary</b>	<b>Application No.</b> 09/775,862	<b>Applicant(s)</b> LIU ET AL.	
	<b>Examiner</b> Ashok B. Patel	<b>Art Unit</b> 2154	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
  - If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
  - If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
  - Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

- 1) ☒ Responsive to communication(s) filed on 30 June 2004.
- 2a) ☒ This action is **FINAL**.                      2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

- 4) ☒ Claim(s) 1-33 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-33 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

**Application Papers**

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All    b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
  2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

- |  |   |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892)  | 4) <input type="checkbox"/> Interview Summary (PTO-413)<br>Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948)                                   | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152)             |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)<br>Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____  |

### DETAILED ACTION

1. Claims 1-33 are subject to examination.

#### ***Response to Arguments***

2. Applicant's arguments filed June 30, 2004 (supplemental reply to office action which includes the reply to office action filed June 18, 2004) have been fully considered but they are not persuasive for the following reasons:

- a. First of all, in response to applicant's arguments against the references, Carcerano and Garvey, individually, one cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981); *In re Merck & Co.*, 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986). The reference Gravey teaches "a method of editing parameter settings for more than one network device where a set of network devices is displayed on a display screen and to edit parameters for a particular network device, the user selects that network device on the display screen using a cursor control device. (col. 2, lines 6 10.) The reference also teaches the buttons in the client process that in response from the client process execute routines to operate on the particular network device based on a submit method associated with the form by stating "to set up additional parameters for the remote access server, the user selects the More button 450 that moves the user along to the second general setup window 500 as illustrated in FIG. 5. The second general setup window 500 allows the user to edit additional parameters that define the operation of the remote access server such as security hosts, a time host, a logging host, designate name servers, and software options. Once

the desired parameter changes have been made, the user selects the OK button 510 to return to the first general setup window 500. (col. 4, lines 52-61).”, thus the buttons associated with the browser display of Garvey to execute routine for setting the parameters on the network devices such that the user has a management tool that allows each user to control, monitor and obtain the status and configuration of a plurality of network devices using a single and preferably familiar interface.

In response to Applicant’s argument that “Carcerano et al.’s claim 16, states that one URL request is used for retrieving data, and a second URL for setting the data. An embodiment of the present application uses a single form (e.g., LJRLI for both getting and setting data” and arguments go on explaining the benefits of having a single form, as stated, the reference Carcerano discloses the two URL requests, one for retrieving data and a second for setting the data. It would have been obvious to one having ordinary skill in the art at the time the invention was made to combine the two URLs in one page or form, since it has been held that forming one piece an article which has formerly been formed in two pieces and put together involves only routine skill in the art. (Howard v. Detroit Stove Works, 150 U.S. 164 (1893).)

**b.** In response to applicant's argument that the references fail to show certain features of applicant's invention, it is noted that the features upon which applicant relies (i.e., “In Carcerano et al., the first URL request identifies a targeted one of the network devices. In an embodiment of the present application, the URL only contains the web server's address, the targeted device information is never visible in the present application's form or URL. Carcerano et al. require constant polling of device information

to keep in sync with their database. In contrast, in an implementation of the present application obtaining device information is event driven. The present application requires less polling of the targeted device, thus reducing the impact to targeted device and network traffic. Carcerano et al. require a database to hold device information. In an implementation of the present specification, an object model in memory is used for holding information. Carcerano et al., first accumulate changes in their database, and then update their device. In the present application, once the object model is modified, the object model may immediately update the device by generating the required commands and sending the required commands to the device.") are not recited in the rejected claim(s). Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988

---

F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993).

**c. Referring to claims 1, 10, 12 and 13:** In response to Applicant's arguments that "Further, claims 1, 10, 12, and 13 recite, associating with the form one of a first submit method for obtaining information about a state of a particular network device, and a second submit method for changing the state of the particular network device. (emphasis added). For antecedents, the "state" which is changed relies on the "state" about which information is changed, and therefore both states are the same "state" information is changed, and therefore both states are the same "state". In contrast, the Office Action relies upon the alleged states associated with the web page portion "Device Status" (125) (which the Office Action implies has "states" such as "printing" or "out of paper") for the state about which information is being obtained and the web page

Art Unit: 2154

portion "Device Features" (127) (which the Office Action implies has "states" such as "12 pages per minute" or "72 MB maximum memory") for the states that are being changed, which are different types of states (assuming that they are "states").", first of all, the Applicant has failed to define "state" in the claims 1, 10, 12 and 13, and as such, the definition of the state was used from Microsoft Press Computer Dictionary, Third Edition, published in 1997 by Microsoft Corporation, which defines the state as "status" being "The condition at a particular time of any of numerous elements of computing – a device, a communications channel, a network station, a program, a bit, or other element- used to report on or to control computer operations.". The reference Carcerano teaches "The management system is interposed between the administrators (or other users) and the network devices. The management system repeatedly polls the network devices for configuration information and maintains a database of this information. When the management system receives a request from a web browser for status (**state**) or configuration information about a network device, the system generates an appropriate response based on the database rather than on information obtained directly from the network device."(col.2, lines 12-21).

Additionally, In response to applicant's arguments regarding "state", Examiner has noticed that the specifications, page 13, line 7-10, do state "Block 260 includes calls to the standard network management methods to obtain the status of one or more network devices, or to reconfigure one or more network devices."

In response to Applicant's arguments and conclusions that "Presumably, the same method is used for performing all the functions associated with the web page of FIG. 7." And "In fact, FIG. 9 includes a "VIEW STATUS/CONFIGURATION OF A DEVICE" (see box 5902) in the same flow diagrams "CHANGE STATUS/CONFIGURATION OF DEVICE" (see box S907), thereby suggesting that both getting information about a device and changing the status or state of a device are performed by the same method.", are erroneous. As stated in the previous office action, there are two different methods as shown in Fig.7, which includes device status (element 125) (a state of a particular network device) and device features (element 127)(changing the state of the particular network device)(associating with the form one of a first submit method for obtaining information about a state of a particular network device, and a second submit method for changing the state of the particular network device; sending the page to the client process;). The reference also teaches that the device features 127 of Fig. 7 can be changed (changing the state of the particular network device) and in response to such changes, browser 83 sends an appropriate URL-encoded request identifying the targeted printer and the updated configuration data by HTTP server 103 runs the CGI script (or ASP web page) identified by the URL in the request so as to update database 105 accordingly. Then, network management server 104 modifies the status or configuration of the device according to the updated database as explained for Fig.7, (col. 12, lines 62-67 and col. 13, lines 1-17) (in response to activation in the client process, executing routines to operate on the particular network device based on a submit method associated with the form.)



**d. Referring to claim 2:** In response to Applicant's argument that "However, the Office Action does not support its contention that inherently by using ASP, values identifying the particular network device are hidden in the form sent from the apparatus for configuring the particular network device to the client. The burden of proof is upon the U.S. Patent and Trademark Office, when alleging that a feature is inherent, and the Applicants respectfully request support for the assertion of inherency.", the reference Soderstrom et al. (US 2001/0047454 A1) in para.[0032] teaches "The server portion also includes a processor 206 and memory 208. The memory 208 includes program code for various processes discussed in the following figures and HTML, .JSP, .ASP or other such web pages to allowing the user to configure individual devices during the configuration phase and to view, open and close files during the operational phase of the server." Thereby it teaches that the ASP framework allows developers to create "ASP" web page files that typically include Visual Basic or Jscript code, as well as other HTML code. The ASP file contains declarations or tags that perform various functions as well as VB script or Jscript code. During operation, the HTTP request specifies the ASP file as the desired resource and, thereafter, the ASP file is used to generate the resulting HTML code in the HTTP response to the client. Furthermore, an ASP file may reference pre-developed or third party server-side library components (e.g., server-side ACTIVEX controls) as well as data bases, such as of the reference Carcerano, or other third party applications to ease a given application programming efforts.

**e. Referring to claim 4-6:** In response to Applicant's argument that "Carcerano et al. or Garvey et al. (whether taken alone or in combination) disclose, teach or suggest using two different methods for obtaining the same information and for tilling in the same plurality of input fields.", the reference Carcerano teaches a system that allows a remote network user to view and update the configuration of network devices by using a web browser on the user's workstation. The system communicates with the web browser using hypertext transfer protocol (HTTP). Requests from the browser are URL-encoded according to HTTP, and are preferably URL-encoded for CGI scripts, ASP web pages, or any other scripts or pages that can be used by the system to dynamically generate responses to the requests. The responses to the web browser's requests preferably are in hypertext markup language (HTML). (col. 1, lines 60-67 and col. 2, lines 1-4). (preparing a page for sending to a client process from a server, the page comprising a form having a plurality of input fields). As stated above, the state is defined as being a status as being an operational indicative for the same device of element 125 and element 126 in Fig. 7, first they are two different methods and , second, as indicated, they do obtain the same information (operational indicative for the same device) in the same plurality of input fields.

**f. Referring to claim 7:** In response to Applicant's arguments that "The Office Action has not shown that the "first environmental variable" is a "Request.ouerystring" and the "second environmental variable" is "Request.Form" .", as stated previously in the office action, the system communicates with the web browser using hypertext transfer protocol (HTTP). Requests from the browser are URL-encoded according to

Art Unit: 2154

HTTP, and are preferably URL-encoded for CGI scripts, ASP web pages, or any other scripts or pages that can be used by the system to dynamically generate responses to the requests. The responses to the web browser's requests preferably are in hypertext markup language (HTML). (col. 1, lines 60-67 and col. 2, lines 1-5). Thereby the reference teaches by the virtue of ASP and HTML, wherein the ASP page that contains a form for data-entry into a database, the user navigates to the page, that is, an HTTP GET causes the server to run the ASP to produce a page for the client. The browser displays the page and the user enters data and submits the form, causing an HTTP POST to the server. The server runs the ASP page again to process the post. The code parses the post parameters, updates the database and produces a new version of the page for the client that informs the user the data has been successfully entered. In the HTML, request.querystring and request.form are conventional ways of passing information to a server page by building a query string into the URL passed to the server and passing user input to the server.

### **Claim Rejections - 35 USC § 103**

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

4. Claims 1-33 are rejected under 35 U.S.C. 103(a) as being unpatentable over Carcerano et al. (hereinafter Carcerano)(US 6,308,205) in view of Garvey et al. (hereinafter Garvey)(US 5,774,667).

**Referring to claims 1 and 2,**

The reference Carcerano teaches a system that allows a remote network user to view and update the configuration of network devices by using a web browser on the user's workstation. The system communicates with the web browser using hypertext transfer protocol (HTTP). Requests from the browser are URL-encoded according to HTTP, and are preferably URL-encoded for CGI scripts, ASP web pages, or any other scripts or pages that can be used by the system to dynamically generate responses to the requests. The responses to the web browser's requests preferably are in hypertext markup language (HTML). (col. 1, lines 60-67 and col. 2, lines 1-4). (preparing a page for sending to a client process from a server, the page comprising a form having a plurality of input fields). As result of the system the web browser interface is displayed as shown in Fig. 7 which includes device status (element 125) (a state of a particular network device) and device features (element 127)(changing the state of the particular network device)(associating with the form one of a first submit method for obtaining information about a state of a particular network device, and a second submit method for changing the state of the particular network device; sending the page to the client process;). The reference also teaches that the device features 127 of Fig. 7 can be changed (changing the state of the particular network device) and in response to such changes, browser 83 sends an appropriate URL-encoded request identifying the

targeted printer and the updated configuration data by HTTP server 103 runs the CGI script (or ASP web page) identified by the URL in the request so as to update database 105 accordingly. Then, network management server 104 modifies the status or configuration of the device according to the updated database as explained for Fig.7, (col. 12, lines 62-67 and col. 13, lines 1-17) (in response to activation in the client process, executing routines to operate on the particular network device based on a submit method associated with the form.) By teaching the use of ASP as indicated above, the reference teaches that ASP provides means for managing the display page state by placing values in hidden form elements where the form elements travel with the form page (the form includes a hidden variable having a value for uniquely identifying the particular network device). The reference fails to teach the submit button in the client process. The reference Garvey teaches a method of editing parameter settings for more than one network device where a set of network devices is displayed on a display screen and to edit parameters for a particular network device, the user selects that network device on the display screen using a cursor control device. (col. 2, lines 6 10.) The reference also teaches the buttons in the client process that in response from the client process execute routines to operate on the particular network device based on a submit method associated with the form by stating "to set up additional parameters for the remote access server, the user selects the More button 450 that moves the user along to the second general setup window 500 as illustrated in FIG. 5. The second general setup window 500 allows the user to edit additional parameters that define the operation of the remote access server such as security hosts, a time host, a logging

Art Unit: 2154

host, designate nameservers, and software options. Once the desired parameter changes have been made, the user selects the OK button 510 to return to the first general setup window 500. (col. 4, lines 52-61). Therefore, it would have been obvious for one having ordinary skill in the art at the time the invention was made to modify Carcerano by adding the buttons associated with the browser display of Garvey to execute routine for setting the parameters on the network devices (executing routines to operate on the particular network device based on a submit method associated with the form) such that the user has a management tool that allows each user to control, monitor and obtain the status and configuration of a plurality of network devices using a single and preferably familiar interface. In addition, this tool should allow many administrators to view and update the configuration of the network devices without generating excessive network traffic that can overburden those devices as taught by Carcerano.

**Referring to claims 3 and 4,**

The reference Carcerano teaches that its management system is interposed between the administrators (or other users) and the network devices. The management system repeatedly polls the network devices for configuration information and maintains a database of this information. When the management system receives a request from a web browser for status or configuration information about a network device, the system generates an appropriate response based on the database rather than on information obtained directly from the network device. Likewise, when the management system receives a request from a web browser to change the status or configuration of a

network device, the system updates the database according to configuration data in the request. The management system then updates the configuration of the network devices according to the updated database. (col. 2, lines 12-26 and Fig. 6 and Fig. 7, col. 12, lines 52-61). (determining whether a first variable for data input associated with the first submit method is empty, and if it is determined that the first variable is not empty, then executing get routines to obtain information about the state of the particular network device, and assigning values for the plurality of input fields based on the information about the state of the particular network device; and wherein the method further comprises sending the page to the client process including the form with the second submit method after said assigning.) The reference also teaches to select the device features from the list of device features, element 127, and the user can change device feature. As stated by the reference, in response to such changes, browser 83 sends an appropriate URL-encoded request identifying the targeted printer and the updated configuration data. As discussed above, HTTP server 103 runs the CGI script (or ASP web page) identified by the URL in the request so as to update database 105 accordingly. Then, network management server 104 modifies the status or configuration of the device according to the updated database. (Fig. 7 and col. 12, lines 62-67 and col. 13, lines 1-17). (obtain the current values of the obtaining current values of the plurality of input fields from a second variable associated with the second submit method; and executing set routines to change the state of the particular network device based on the current values of the plurality of input fields.)

**Claims 5, 6 and 7,**

The reference Carcerano teaches a system that allows a remote network user to view and update the configuration of network devices by using a web browser on the user's workstation. The system communicates with the web browser using hypertext transfer protocol (HTTP). Requests from the browser are URL-encoded according to HTTP, and are preferably URL-encoded for CGI scripts, ASP web pages, or any other scripts or pages that can be used by the system to dynamically generate responses to the requests. The responses to the web browser's requests preferably are in hypertext markup language (HTML). (col. 1, lines 60-67 and col. 2, lines 1-5). Thereby the reference teaches by the virtue of ASP and HTML, wherein the ASP page that contains a form for data-entry into a database, the user navigates to the page, that is, an HTTP GET causes the server to run the ASP to produce a page for the client. The browser displays the page and the user enters data and submits the form, causing an HTTP POST to the server. The server runs the ASP page again to process the post. The code parses the post parameters, updates the database and produces a new version of the page for the client that informs the user the data has been successfully entered. In the HTML, request.querystring and request.form are conventional ways of passing information to a server page by building a query string into the URL passed to the server and passing user input to the server.

**Referring to claims 8 and 9,**

The reference Carcerano teaches a system that allows a remote network user to view and update the configuration of network devices by using a web browser on the user's workstation. The system communicates with the web browser using hypertext transfer



protocol (HTTP). Requests from the browser are URL-encoded according to HTTP, and are preferably URL-encoded for CGI scripts, ASP web pages, or any other scripts or pages that can be used by the system to dynamically generate responses to the requests. The responses to the web browser's requests preferably are in hypertext markup language (HTML). (col. 1, lines 60-67 and col. 2, lines 1-5). The reference Carcerano teaches the processor that is configured for preparing a page and, executing routines are performed by the processor configured based on statements of a scripting language in a single script file. (Fig. 4, elements 91 and 106). The reference also teaches that if a URL-encoded request includes changes to the status or configuration of a device on network 1, the CGI script called by HTTP server 103 in response to that request enters those changes into database 105 which in turn changes the state of the network device. (col. 9, lines 38-42). The reference fails to teach the submit button in the client process. The reference Garvey teaches a method of editing parameter settings for more than one network device where a set of network devices is displayed on a display screen and to edit parameters for a particular network device, the user selects that network device on the display screen using a cursor control device. (col. 2, lines 6 10.) The reference also teaches the buttons in the client process that in response from the client process execute routines to operate on the particular network device based on a submit method associated with the form by stating "to set up additional parameters for the remote access server, the user selects the More button 450 that moves the user along to the second general setup window 500 as illustrated in FIG. 5. The second general setup window 500 allows the user to edit additional

parameters that define the operation of the remote access server such as security hosts, a time host, a logging host, designate nameservers, and software options. Once the desired parameter changes have been made, the user selects the OK button 510 to return to the first general setup window 500. (col. 4, lines 52-61). Therefore, it would have been obvious for one having ordinary skill in the art at the time the invention was made to modify Carcerano by adding the buttons associated with the browser display of Garvey to execute routine for setting the parameters on the network devices such that the user has a management tool that allows each user to control, monitor and obtain the status and configuration of a plurality of network devices using a single and preferably familiar interface. In addition, this tool should allow many administrators to view and update the configuration of the network devices without generating excessive network traffic that can overburden those devices as taught by Carcerano.

**Referring to claims 10 , 11 and 17,**

Claims 10,11 and 17 are claims to a computer-readable medium carrying one or more sequences of instructions for configuring network devices, which instructions, when executed by one or more processors, cause the one or more processors to carry out the steps of method of claims 1, 2 and 3. Therefore, claims 10,11 and 17 are rejected for the reasons set forth for claims 1, 2, 3 and 4 and, the reference Carcerano's teaching of processor (Fig. 4, element 91) executing the instructions.

**Referring to claims 12 and 14,**

Claims 12 and 14 are claims to an apparatus for configuring network devices with a network interface; and one or more processors connected to the network interface, the

Art Unit: 2154

one or more processors configured for carrying out the steps of method of claim 1. Therefore, claims 12 and 14 are rejected for the reasons set forth for claim 1 and the reference Carcerano's teaching of processor (Fig. 4, element 91) and the network interface (Fig. 4, element 47).

**Referring to claim 13,**

Claim 13 is a claim to an apparatus for configuring network devices which provides the means to carry out the steps of method of claim 1. Therefore, claim 13 is rejected for the reasons set forth for claim 1.

**Referring to claims 15 and 33,**

The reference Carcerano teaches the second submit method and an apparatus wherein the second submit method includes at least includes changing the state of the particular network device; and obtaining the information about the state of the particular network device as stated in Fig.7. for the reasons stated in claim 1.

**Referring to claim 16,**

The reference Carcerano teaches a system that allows a remote network user to view and update the configuration of network devices by using a web browser on the user's workstation. The system communicates with the web browser using hypertext transfer protocol (HTTP). Requests from the browser are URL-encoded according to HTTP, and are preferably URL-encoded for CGI scripts, ASP web pages, or any other scripts or pages that can be used by the system to dynamically generate responses to the requests. The responses to the web browser's requests preferably are in hypertext markup language (HTML). (col. 1, lines 60-67 and col. 2, lines 1-4). (a client process

receiving from a server, a page comprising a form having a plurality of input fields and a submit button; .receiving user input;). As result of the system the web browser interface is displayed as shown in Fig. 7 which includes device status (element 125) (a state of a particular network device) and device features (element 127)(changing the state of the particular network device)( wherein the form is associated with one of a first submit method for obtaining the information about a state of a particular network device, and a second submit method for changing the state of the particular network device, based on the user input; and wherein the information is obtained by executing routines to operate on the particular network device based on a submit method associated with the form.). The reference also teaches that the device features 127 of Fig. 7 can be changed (changing the state of the particular network device) and in response to such changes,

browser 83 sends an appropriate URL-encoded request identifying the targeted printer and the updated configuration data by HTTP server 103 runs the CGI script (or ASP web page) identified by the URL in the request so as to update database 105 accordingly. Then, network management server 104 modifies the status or configuration of the device according to the updated database as explained for Fig.7, (col. 12, lines 62-67 and col. 13, lines 1-17) in response to the activating, sending a signal associated with the user input to the server; the client process receiving from the server the page having information in the plurality of input fields, executing routines to operate on the particular network device based on a submit method associated with the form.) By teaching the use of ASP as indicated above, the reference teaches that ASP provides means for managing the display page state by placing values in hidden form

elements where the form elements travel with the form page (the form includes a hidden variable having a value for uniquely identifying the particular network device). The reference fails to teach the submit button in the client process. The reference Garvey teaches a method of editing parameter settings for more than one network device where a set of network devices is displayed on a display screen and to edit parameters for a particular network device, the user selects that network device on the display screen using a cursor control device. (col. 2, lines 6-10.) The reference also teaches the buttons in the client process that in response from the client process execute routines to operate on the particular network device based on a submit method associated with the form by stating "to set up additional parameters for the remote access server, the user selects the More button 450 that moves the user along to the second general setup window 500 as illustrated in FIG. 5. The second general setup window 500 allows the user to edit additional parameters that define the operation of the remote access server such as security hosts, a time host, a logging host, designate nameservers, and software options. Once the desired parameter changes have been made, the user selects the OK button 510 to return to the first general setup window 500. (col. 4, lines 52-61). Therefore, it would have been obvious for one having ordinary skill in the art at the time the invention was made to modify Carcerano by adding the buttons associated with the browser display of Garvey to execute routine for setting the parameters on the network devices (executing routines to operate on the particular network device based on a submit method associated with the form) such that the user has a management tool that allows each user to control, monitor and obtain the status and configuration of

a plurality of network devices using a single and preferably familiar interface. In addition, this tool should allow many administrators to view and update the configuration of the network devices without generating excessive network traffic that can overburden those devices as taught by Carcerano.

**Referring to claims 18-23,**

Claims 18-23 are claims to a computer-readable medium carrying one or more sequences of instructions for configuring network devices, which instructions, when executed by one or more processors, cause the one or more processors to carry out the steps of method of claims 4-9. Therefore, claims 18-23 are rejected for the reasons set forth for claims claims 4-9.

**Referring to claim 24,**

Claim 24 is a claim to a computer-readable medium carrying one or more sequences of instructions for configuring network devices, which instructions, when executed by one or more processors, cause the one or more processors to carry out the steps of method of claim 15. Therefore, claim 24 is rejected for the reasons set forth for claim 15.

**Referring to claims 25-32,**

Claims 25-32 are claims to an apparatus for configuring network devices with a network interface; and one or more processors connected to the network interface, the one or more processors configured for carrying out the steps of method of claims 2-9. Therefore, claims 25-32 are rejected for the reasons set forth for claim 2-9.

***Conclusion***

**5. THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

---


Any inquiry concerning this communication or earlier communications from the examiner should be directed to Ashok B. Patel whose telephone number is (703) 305-2655. The examiner can normally be reached on 8:00am-5:00pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John A Follansbee can be reached on (703) 305-8498. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Art Unit: 2154

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Abp  
\*\*\*



JOHN FOLLANSBEE  
SUPERVISORY PATENT EXAMINER  
TECHNOLOGY CENTER 2100

---